

Multiplex Communication with Synchronous Shift and Weight Learning in 2D Mesh Neural Network

Takuya Kamimura¹, Yasushi Yagi², Shinichi Tamura¹, Yen-Wei Chen³

¹NBL Technovator Co., Ltd. Shindachimakino, Sennan, Japan

²ISIR, Osaka University, Mihogaoka, Ibaraki City, Osaka, Japan

³Ritsumeikan University, Nojihigashi, Kusatsu-shi, Shiga, Japan

Email address:

kamimura@nbl-technovator.jp (T. Kamimura), yagi@sanken.osaka-u.ac.jp (Y. Yagi), tamuras@nblmt.jp (S. Tamura),

chen@is.ritsumeikan.ac.jp (Yen-Wei C.)

To cite this article:

Takuya Kamimura, Yasushi Yagi, Shinichi Tamura, Yen-Wei Chen. Multiplex Communication with Synchronous Shift and Weight Learning in 2D Mesh Neural Network. *Automation, Control and Intelligent Systems*. Vol. 3, No. 5, 2015, pp. 63-70. doi: 10.11648/j.acis.20150305.11

Abstract: We have previously proposed a multiplex communication system in a neural network. However, this system is designed to force the network to communicate in a multiplexed manner, in which “codes” or “temporal sequences” are inevitably induced. This means that the network has a main loop and coding/decoding circuits, which are somewhat artificial. In this paper, we show that it is also possible to communicate without these artificial guidance aids by multiplexing in a 2D mesh-type neural network, where learning procedures are used to find paths from an originating neuron to a destination neuron. We also provide statistics from these neural networks to show that random sequences occur more frequently than non-random sequences.

Keywords: Brain Information Processing, Neural Circuit, Pseudo-Random Sequence, M-Sequence, Multiplex Communication

1. Introduction

Investigation of the manner in which information is communicated in the brain is an important theme in brain science. We have previously developed a time-shift map for analysis of the transmission of electroencephalogram (EEG) or magnetoencephalogram (MEG) signals in the brain [1], and have shown that it is effective for diagnosis of transient global amnesia (TGA) [2]. The time-shift map is a graphical representation that shows the time difference at which the cross-correlation between a brain wave on a point and that at another point becomes a maximum. An example of a time-shift map is shown in Fig. 1. The advantageous feature of this method when compared with the magnetic resonance imaging (MRI) dipole diagram [3] is that this method can follow even small signal flows. We can see that sub-tasks are processed in each hemisphere within 5 ms, and after 10 ms, the results are exchanged between the hemispheres. The question that arises here is how the neuron cells locate the target cells to send the required signal or, alternatively, how the responsible cells can obtain the necessary signals from the due cells, even when they are at remote locations. This is a problem of finding communication links in a neuronal

network. This paper will contribute to a solution to this question, and provide a basic idea of how events are stored in the brain and how the brain can associate/recall matters related to these events.

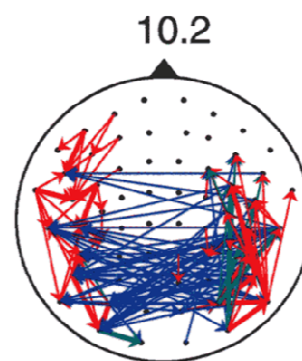


Fig. 1. Time-shift map of 10.2 Hz wave from MEG. Red < 5 ms < Green < 10 ms < Blue.

Figure 2 shows the simplest possible code multiplex communication of 2×2 communication on a 2D synchronous shift neural network, i.e., the number of input neurons is two and the number of output neurons is also two. Expanding upon this, we have previously proposed a multiplexed

communication model on a 2D synchronous shift neural network that is composed of a main loop circuit and some coding/decoding neural networks [4], as shown in Fig. 3. The network is forced to communicate in a multiplexed manner, because there are only two lines between the three transmitting neurons and the three receiving neurons of the three communication channels. It was then observed that the network frequently generates and uses pseudo-random sequences, including M-sequences [5]. It is well known that in many real-world artificial communication systems, such as mobile phones, pseudo-random sequences—and particularly M-sequences—are used as almost orthogonal codes that are easy to discriminate from each other. Additionally, we have observed the M-sequence family occurring in spike trains from cultured neural networks significantly more frequently than would occur by chance [6, 7].

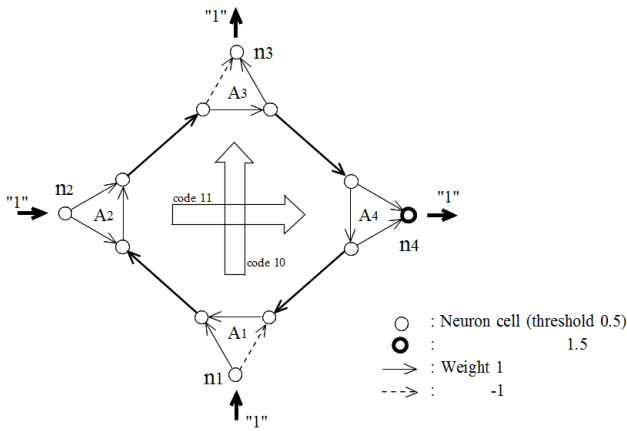


Fig. 2. Simplest example of a 2×2 multiplex communication channel by a neural network. Each neuron expressed by a small circle outputs a “1” if the weighted sum of inputs exceeds the required threshold, and outputs a “0” otherwise. If “1” is input to n_1 , then it is converted into a sequence “10” by the local loop A_1 and is transferred to A_2 and A_3 . Finally, “1” is output from n_3 . Similarly, an input “1” to n_2 is transferred to A_4 , and a “1” is output from n_4 . This means that there are two independent communication channels working by temporal code multiplexing on the loop line.

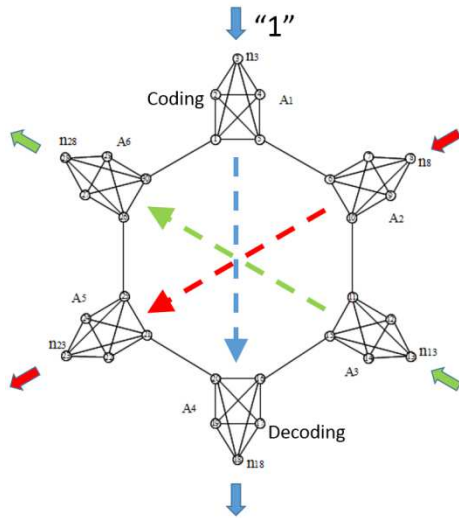


Fig. 3. A 3×3 communication model. The pulse “1” sent from neuron n_3 is supposed to be received by n_{18} only, while that sent from n_8 is supposed to be received by n_{23} only, and that sent by n_{13} is supposed to be received by n_{28} only.

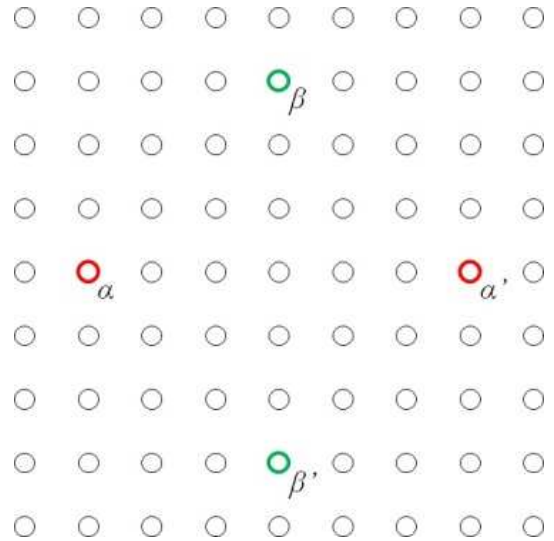
However, it remains to be seen whether it is possible to communicate in a multiplexing manner in more natural neural networks without using an artificial shape to force multiplexing. Therefore, in this paper, we treat a more naturally homogeneous grid-shaped neural network. While our previous network was tested by generating numerous random networks without a learning function, here we provide a learning function for the networks to enhance the communication links.

2. Path Search Model

It is well known that a neuronal cell has more than 1000 synapses [8], and that large numbers of connection paths and loops thus exist in a neural network [9–11]. Stimulation is transmitted from one neuronal cell to another through the synapses, and the membrane electrical potential is thus increased. If the electrical potential exceeds a specific threshold, then the neuronal cell is fired, and if the potential is beneath the threshold, then the cell is not fired. The firing model [12] of a neuronal cell is as shown in Eq. (1).

$$E(t) = \begin{cases} 1 & \sum_{n=1}^N E_n(t-1)W_{E,E_n} > \text{threshold} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Here, $E(t)$ represents the state of cell E at a discrete time t (where $t=0, 1, 2, \dots$), 1 represents the firing state, and 0 represents the static state. N is the number of other cells that are connected to the cell, and W_{E,E_n} is the connection weight between E and E_n . The state of a cell is thus determined on the basis of whether the total sum of the products of the connection weight and the state of the other connected cell at the previous clock cycle time exceeds a threshold or not. In each neuron cell, the state will change with progress of t . We regard the communication as successful if an output cell is fired when external stimulation is provided to an input neuron cell and the stimulation is thus transmitted to the output cell through the neuron cells in the neural network.



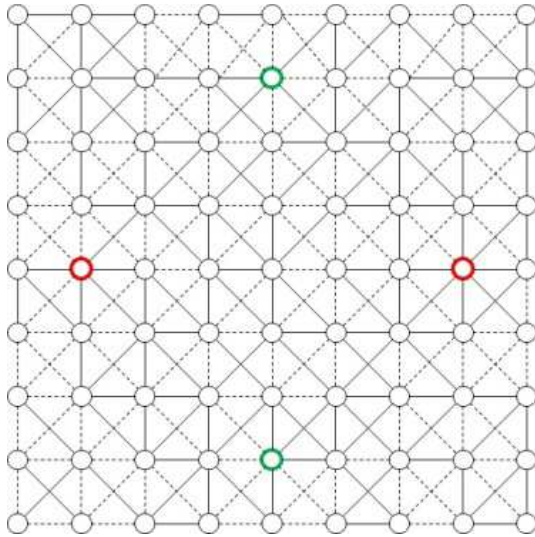
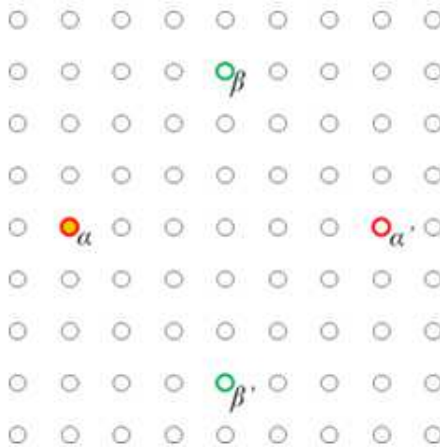
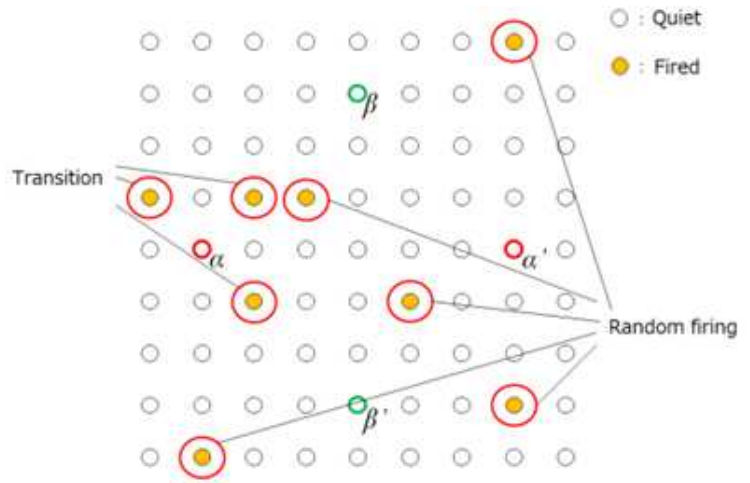


Fig. 5. Bidirectional connection weights. Solid lines are positive weight connections, and dotted lines are negative weight connections.

We formed a 2×2 multiplexing communication model with two input neuron cells to transmit the information and two corresponding receiving neuron cells on a 9×9 grid-shaped



(a) $t=0$.



(b) $t=1$.

Fig. 6. Example of a state transition.

2.1. Updating Weights

The weight set between cells expresses the transmission efficiency of the relevant synapse. Synapse transmission efficiency in a natural neuronal network is reinforced if a post-synaptic neuron is fired by the firing of a presynaptic neuron or even by chance. This is called the Hebb rule [14]. In the path search model, the connection weights are also updated based on the Hebb rule. In this case, there are two cases of the weight updating rule of reinforcement by simultaneous firing and that based on the state one clock cycle before firing. These cases are expressed using Eq. (2) and Eq. (3), respectively.

$$\Delta w_i = \alpha s_i(t) s_{i+1}(t) + \beta_1 w_{i-1} + \beta_2 w_{i+1} \quad (2)$$

neural network. We call this the “path search model” [13]; the model is as shown in Fig. 5. Here, α and β are the input cells for information transmission, and α' and β' are the cells that receive information from α and β , respectively. Between a specific cell and its eight neighbouring cells, the connection weights are set randomly at time 0 within the range $[-1, 1]$. An example of the initial weights is shown in Fig. 5. Here, a solid line represents a positive weight connection, and a broken line represents a negative weight connection. Firing information input to input cells α and β is transmitted in the network, with the transitions of the cell states occurring according to the connection weights that are set between the cells and Eq. (1) with the progress of the time steps.

In addition, cells are sometimes fired voluntarily and randomly, irrespective of the firing transition. We call this phenomenon “random firing.” With the combination of firing by the state transition and random firing, the firing cells thus move.

An example of a state transition is shown in Fig. 6. We can see that firing in Fig. 6(a) of α at time $t=0$ leads to the change into Fig. 6(b) at time $t=1$.

$$\Delta w_i = \alpha s_i(t) s_{i+1}(t-1) + \beta_1 w_{i-1} + \beta_2 w_{i+1} \quad (3)$$

Here, w_i is the attention paid weight, $s_i(t)$ is the state of cell i at time t , and $i+1$ represents the cell that is connected to cell i with weight w_i . Weights w_{i-1} and w_{i+1} represent those weights ahead of the extending weight along the connected direction, and have the role of an inertia term to ensure that the weight updates smoothly. In addition, α , β_1 , and β_2 are coefficient parameters. Figure 7 shows a schematic diagram of these weights.

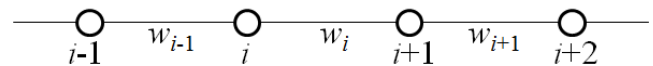


Fig. 7. Schematic diagram of weights.

In the case of simultaneous firing reinforcement, as shown in Eq. (2), the weights are updated by referring to the states of cell i and cell $i+1$ at time t , and in the case of firing reinforcement one clock cycle before firing, the weights are updated by referring to the state of cell i at time t and the state of cell $i+1$ at time $t-1$, respectively. The increment of Δw_i by a weight update is divided equally among the remaining neighbours, and the same amount is deducted from each of the weights in the neighbours. Therefore, the total sum of weights of the whole model does not change, even after the weights are updated. While we proposed two updating rules, there were no differences in the results produced by these rules, and thus we simply applied Eq. (2) in the following experiments.

2.2. Success or Failure of Communication

In a succession of state transitions, a case occurs where the firing information of the transmitting side (α , β) is transmitted to the receiving side (α' , β') and the neurons are fired. In this case, we can regard the information as being transmitted and we can also regard the communication as being successful.

In case of a 2×2 multiplex communication, the information from α should be transferred to α' , while that of β should be transferred to β' . Therefore, when the information is input to α at $t=0$, the number of firings in the range from $t=6$ to $t=63$ should be $N(\alpha') > N(\beta')$, and when the information is input to β at $t=0$, then the number of firings of β' should be $N(\alpha') < N(\beta')$, and only in a case such as this can we regard the communication as having succeeded fully. Here, $N(\alpha')$ represents the number of firings of α' within the corresponding period (i.e., from $t=6$ to $t=63$, or from $t=6$ to $t=31$).

2.3. Sequence That Appeared at the Cell

When we observe all 81 cells of the path search model and pay attention to each cell, we can obtain a sequence in which 0 and 1 appear with the progress of the time step t . Then, we can obtain knowledge of what contributes to the 2×2 communication process by observing and analysing these sequences.

3. Experimental and Discussion

We performed a simulation experiment for 2×2 communication using the path search model.

3.1. Experiment 1

We performed the experiment in three steps (①–③), composed of the weight learning term, an α test, and a β test, as shown in Fig. 8.

① Weight learning term: We perform up to $t=63$ path searches when setting the state of α in Fig. 8 as 1 at $t=0$. In other words, we update the connection weights based on the learning rule.

② α test: Using a model after the weight learning process, we set the state of α in Fig. 8 to be 1 at time $t=0$ and watch the state transition up to $t=63$. During this process, we do not update the weights. Here, if $N(\alpha') > N(\beta')$, then the α test is

successful, while if $N(\beta') < N(\alpha')$, we regard the α test as a failure.

③ β test: Using a model after the weight learning process, we set the state of β in Fig. 5 to be 1 at the time $t=0$ and watch the state transition to $t=63$. During this process, we again do not update the weights. Here, if $N(\beta') > N(\alpha')$, then the β test is successful, and if $N(\alpha') < N(\beta')$, then we regard the β test as a failure.

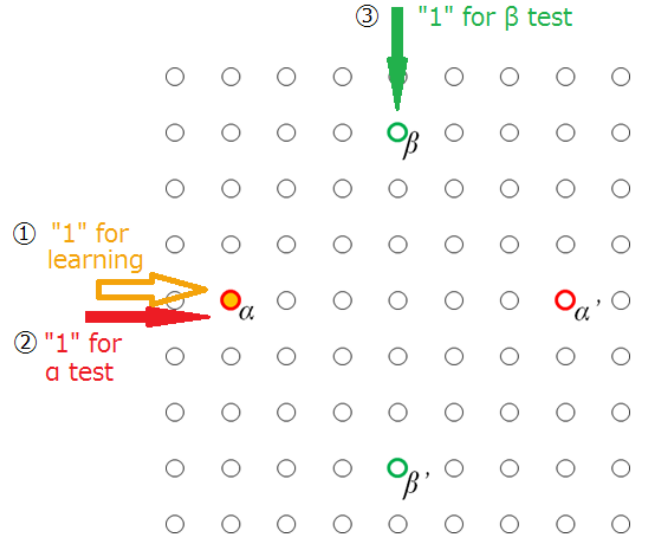


Fig. 8. Experiment 1: The firing state shown is that at $t=0$ of learning phase ①, where α is stimulated. We tried 10^6 times of a series of phase ①–③ tests.

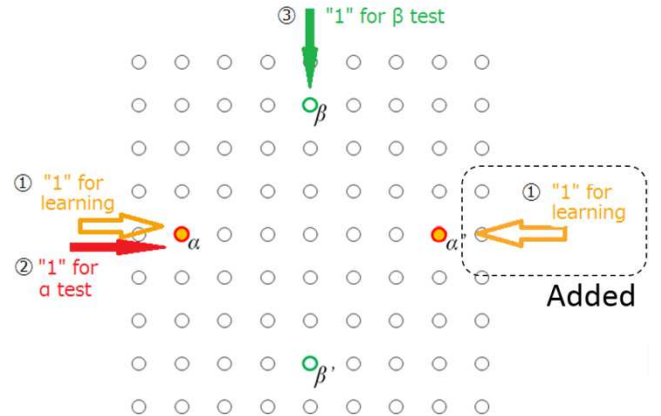


Fig. 9. Experiment 2: Bidirectional learning. The firing state shown is that at $t=0$ of learning phase ①, where α and α' are both stimulated. We tried 10^6 times of a series of phase ①–③ tests.

3.2. Experiment 2

In each weight learning term, weight updates were only performed based on the transition from α in Experiment 1, as shown in Fig. 8. Therefore, we performed weight updates based on the information from the transmission side only, and did not have the information from the cell on the reception side that was the goal for the information. In Experiment 2, as an improvement to the learning process, we formulated a bidirectional learning process, where the states of both α and α' were set as “1” at time $t=0$, and we then performed a path search up to $t=63$, as shown in Fig. 9.

3.3. Experiment 3

In the weight learning processes of Experiments 1 and 2, the learning was based solely on the information of the α side and did not have access to the information of the β side, as shown in Figs. 8 and 9. Therefore, in Experiment 3, we used both-channel learning, as shown in Fig. 10. In this case, both states α and α' are set to be “1” at time $t=0$ and the path search is then performed up to $t=31$. After that, we cleared all the states and set the states of β and β' to be “1” at time $t=0$ and again performed the path search up to $t=31$.

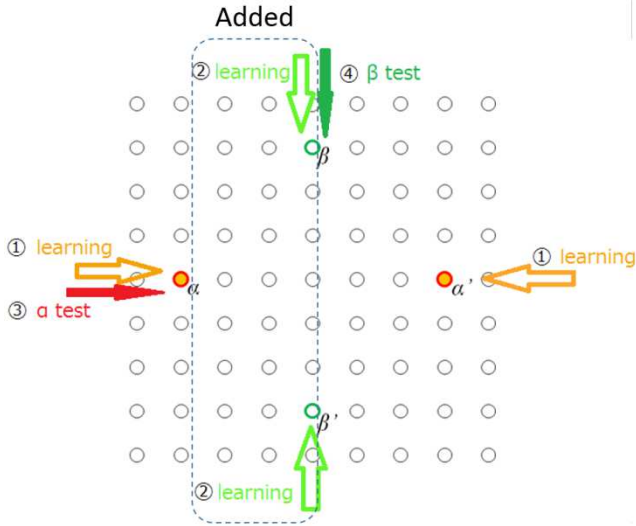


Fig. 10. Experiment 3: Both-channel learning. The firing state shown is that at $t=0$ of learning phase ①, where α and α' are both stimulated. We tried 10^6 times of a series of phase ①-③ tests.

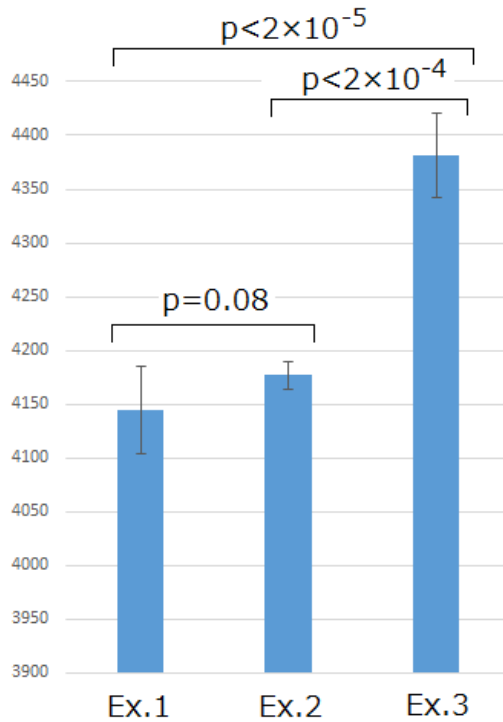


Fig. 11. Average numbers of full successes among 10^6 trials. Each experiment is repeated five times and the standard deviations are also shown as error bars.

The results of Experiments 1, 2, and 3 are shown in Fig. 11. Here, full success means the case where both the α test and the β test succeeded. From each experiment, we can see that 2×2 communication can be realized in the path search model based on randomly assigned initial weights.

In addition, learning becomes easier to advance by using not only the information from the transmission side but also the information from the reception side, and we see that the number of successes increases from a comparison of Experiments 1 and 2, although it is not significant in the 5% level but is significant in the 10% level. We can also see from Experiment 3 that the number of 2×2 communication successes definitely increases by learning in both the α and β channels sequentially when compared with single channel learning in only α or β .

3.4. Experiment 4

In Experiment 3, by varying the parameters of the random firing rate or the firing threshold of the cell, and by performing reinforcement by simultaneous firing or reinforcement by a single clock period before firing, we observed the types of sequences that appeared most often when the 2×2 communication was successful or when it failed. The parameters that we varied were two kinds of random firing rates of 0.1 and 0.05, four kinds of firing thresholds of 0, 0.25, 0.5, and 0.6, and two kinds of reinforcement by simultaneous firing or by a single clock period before firing.

The target sequences to be checked are the sequences with “1” at both ends and with lengths ranging 3 to 8 digits. This gives a total of 120 kinds of target sequence, as follows:

Length 3: [1 1 1] of one kind;

Length 4: [1 0 1 1], [1 1 0 1], [1 1 1 1] of three kinds;

Length 5: [1 0 0 1 1], [1 0 1 0 1], ...[1 1 1 1 1] of seven kinds;

Length 6: [1 0 0 0 1 1], [1 0 0 1 0 1], ...[1 1 1 1 1 1] of 15 kinds;

Length 7: [1 0 0 0 0 1 1], [1 0 0 0 1 0 1], ...[1 1 1 1 1 1 1] of 31 kinds;

Length 8: [1 0 0 0 0 0 1 1], [1 0 0 0 0 1 0 1], ...[1 1 1 1 1 1 1 1] of 63 kinds.

We classify these 120 kinds of sequences into two kinds of sequences: one where the arrangement of “0s” and “1s” is a random-type arrangement and another where the arrangement is not random, as follows.

Case 1: Six output sequences composed of all “1s”, i.e., [1 1 1], [1 1 1 1], ..., [1 1 1 1 1 1 1 1] are regarded as non-random sequences. All others are regarded as random sequences.

Case 2: In addition to the six kinds of non-random sequences composed of all “1s”, the three sequences of [1 0 1 0 1], [1 0 0 1 0 0 1], [1 0 1 0 1 0 1] are also regarded as non-random sequences. That is, totally nine sequences are regarded as non-random sequences. The three sequences of [1 0 1 0 1], [1 0 0 1 0 0 1], [1 0 1 0 1 0 1] may be called “semi-random” sequences.

For these output sequences, we compare the numbers of their appearances when the communications succeeded and failed. We divide the number of appearances when successful

by the number of the appearances when the communication failed. Therefore, if these values exceed 1, then the sequence is relatively frequent when the communication is successful, and if the value is less than 1, the sequence is relatively frequent when the communication failed. We show the average rates of the detected sequences in the results for all parameters in Fig. 12 on the upper side, with an enlarged section on the lower side.

From Fig. 12, we can see that the perfectly non-random

sequences of $[1\ 1\ 1]$, $[1\ 1\ 1\ 1]$, and $[1\ 1\ 1\ 1\ 1]$ with all “1” outputs had values of less than 1, while the other sequences tended to exceed 1. This result shows that when the 2×2 communication succeeded, random-like sequences appeared relatively frequently, and when the communication failed, the non-random sequences appeared relatively frequently. Therefore, this indicates that the random-like sequences contribute to the success of the communication.

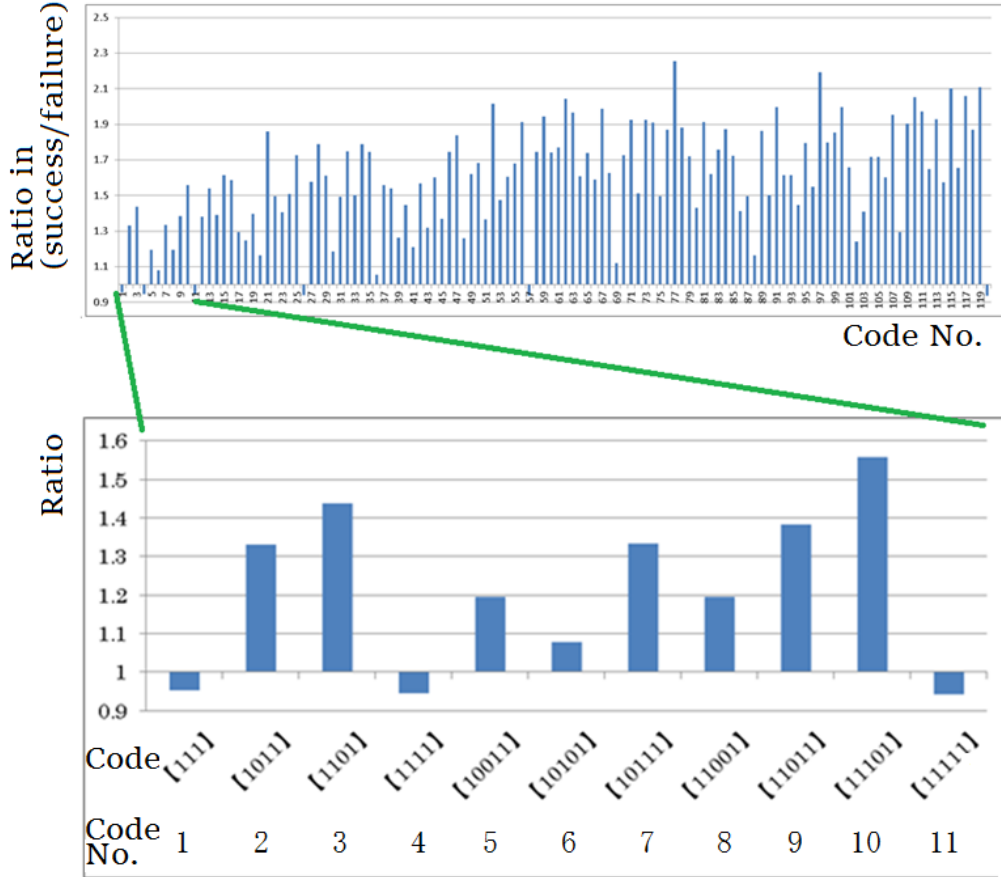


Fig. 12. Ratio of number of sequences in the cases of success/failure that appeared in the output trains for 120 kinds of sequences.

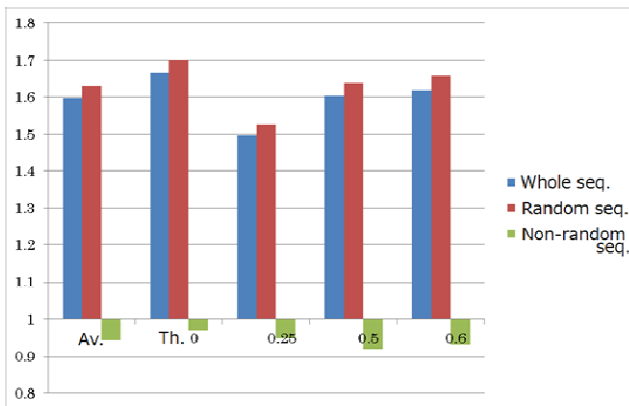


Fig. 13. Comparison of the ratios of numbers of sequences in the cases of success/failure that appeared in the output trains when varying the threshold (Th.). Av. means average. The semi-random sequences are regarded as random sequences in this case.

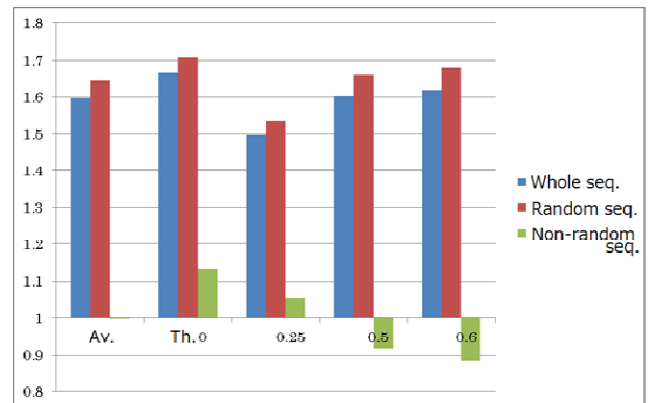


Fig. 14. Comparison of the ratios of numbers of sequences in the cases of success/failure that appeared in the output trains when varying the threshold. The semi-random sequences are regarded as non-random sequences in this case.

In Fig. 13, the output ratios of complete sequences, random sequences, and non-random sequences for various values of the threshold are shown. In this figure, the non-random sequences are again those composed of all “1s” (Case 1). We can see from Fig. 13 that, irrespective of the firing threshold, the random sequences appear more often when the communication succeeded than when communication failed.

In Fig. 14, the results with regard to the nine kinds of non-random sequence, including the three kinds of semi-random sequence of [1 0 1 0 1], [1 0 0 1 0 0 1], [1 0 1 0 1 0 1], are shown (Case 2).

In addition, we know from the results shown in Fig. 14 that a difference in the ratio of the number of output random sequences and non-random sequences observed in the cases of communication success and failure increases when we increase the threshold. This indicates that when the firing threshold of the cell rises, the quantity of firing decreases, and the degree of difficulty of the information transmission and communication processes increases. In such situation, the random characteristics of the sequences became more important.

As a result of the experiments, the factor that is most important to the success or failure of the communication is known to be the random characteristics of a sequence output by a cell in a path search model.

3.5. Synchronous Lateral Inhibition-Type Reception

We have confirmed in a previous loop-type multiplex communication neural network that when we know the time at which the signal is due to arrive at the receiving neuron, the lateral inhibition (LI)-type receptive field characteristic [15] was effective and its recognition rate was improved considerably [4]. This was a case where the network shape was fixed and the process was performed without learning. We also performed an experiment in the present neural network to receive a pulse train with LI-type receptive field under the assumption that the pulse train will arrive with the lowest possible delay. Because the pulse takes at least six clock cycles to arrive and never arrives at time 5 or earlier, the shape of the LI-type receptive field is set to be asymmetric, unlike the ordinary symmetric field, as shown in Fig. 15. We can see from Fig. 16 that the correct recognition rate is improved by about six times when compared with the results of Experiments 1, 2 and 3. The LI-type receptive field has characteristics in common with the living body neural network and this is also shown in this experiment, although it is assumed that the arrival time can be anticipated beforehand by the receptive neuron. This temporal anticipation ability contributes to the improved correct reception rate. Then, LI reception is classified as synchronous reception, while the previous simple counting reception methods of Experiments 1, 2, and 3 can capture the signals irrespective of the time, and are thus called asynchronous reception methods.

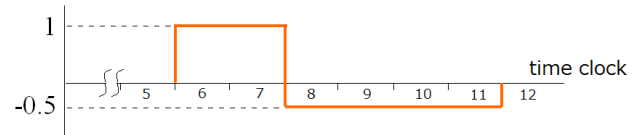


Fig. 15. Weight of LI-type temporal receptive field.

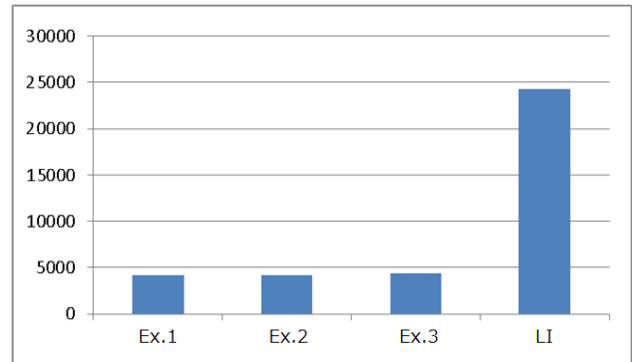


Fig. 16. Numbers of full successes by LI-type temporal receptive field (LI: synchronous reception) among 10^6 trials shown for comparison with the results (Experiments 1, 2, 3; asynchronous reception) of Fig. 11.

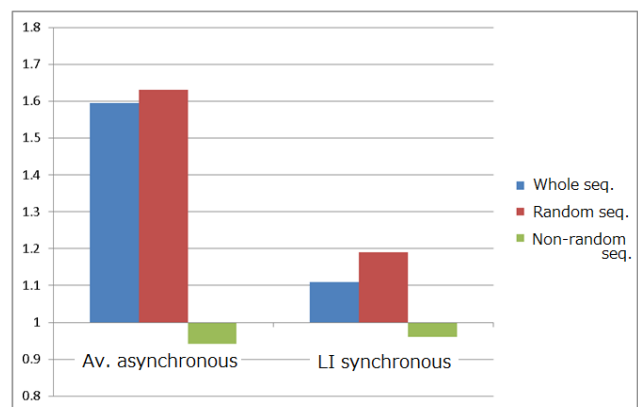


Fig. 17. Comparison of the ratios for success and failure of the number of sequences in pulse trains on average by asynchronous reception from Fig. 11 vs. that by LI synchronous reception.

We can see from Fig. 17 that in the LI reception case, while the ratios of the random and non-random sequences for success and failure were both close to 1, the difference between the cases of success and failure became small. However, the relative significance of the random sequences in LI reception seems to have increased. In this case, the three kinds of semi-random sequences are included in the non-random sequence (Case 2).

4. Discussion and Conclusions

In a previous paper, we proposed multiplex communication systems in neural networks. However, these systems are composed and designed such that they force the network to communicate in a multiplexed manner where “codes” or “temporal sequences” are inevitably induced. In this case, the network has a main loop and some coding/decoding circuits, and these aspects are somewhat artificial. In this paper, we have shown that without these artificial guidance aspects, it is

also possible to communicate via multiplexing on a 2D mesh-type neural network, where some learning procedures are used to find the paths from an originating neuron to a destination neuron.

In this study, we built a path search model on a homogeneous mesh-type 2D neural network with a synchronous state transition; it was shown in the model that 2×2 multiplex communication was possible. In addition, given the possibility of multiplex communication, we knew that the random nature of the sequence of pulses output by a cell had an important meaning. Specifically, we found that multiplexing communication often failed when the network was observed to have many cells that produced less random firing phenomena (e.g., sequences such as “1111”), and, in contrast, we often succeeded in multiplex communication when the network was observed to have many cells that produced richly random firing phenomena (e.g., sequences such as “1101”).

From these results, we knew that random sequences or strongly random sequences greatly enhanced the possibility of communication using methods such as time and space multiplexed communication, as well as artificial communication systems. We also found that when we can anticipate the signal arrival by LI-type synchronous reception, the communication success rate is greatly improved.

While we could confirm the possibility of multiplexed communication on even homogeneous neural networks as a more difficult scheme, our scheme is still limited in that there are only two communication channels and the absolute values of the correct recognition rate are less than 0.5%, despite the addition of the learning ability to the proposed scheme. These aspects should be improved as the next stage of our work. Additionally, although the state transitions of cells in the present paper are synchronous throughout the network, the work should be extended to the asynchronous actions of the cells to investigate the intelligent functions of a natural neuronal network. These aspects will be addressed hereafter in our series of papers.

Acknowledgements

The authors would like to thank Dr. Yoshi Nishitani of Osaka University for his helpful suggestions.

This work was supported in part by Grants-in-Aid for Scientific Research of Exploratory Research (21656100, 25630176) and Scientific Research (A) (22246054) from the Japan Society for the Promotion of Science.

The authors declare that there is no conflict of interest regarding the publication of this paper

References

- [1] Yuko Matsumoto-Mizuno, Kozo Okazaki, Amami Kato, Toshiki Yoshimine, Yoshinobu Sato, Shinichi Tamura, and Toru Hayakawa: “Visualization of epileptogenic phenomena using crosscorrelation analysis: Localization of epileptic foci and propagation of epileptiform discharges,” *IEEE Transactions on Biomedical Engineering*, Vol. 46, No. 3, pp. 271–279, 1999.
- [2] Yuko Mizuno-Matsumoto, Masatsugu Ishijima, Kazuhiro Shinosaki, Takashi Nishikawa, Satoshi Ukai, Yoshitaka Ikejiri, Yoshitsugu Nakagawa, Ryouhei Ishii, Hiromasa Tokunaga, Shinichi Tamura, Susumu Date, Tsuyoshi Inouye, Shinji Shimojo, Masatoshi Takeda: “Transient Global Amnesia (TGA) in an MEG Study,” *Brain Topography*, Vol.13, No.4, pp.269-274, 2001.
- [3] Peter C Hansen, Morten L Kringelbach, Riitta Salmelin, *MEG: An Introduction to Methods*, Oxford University Press Inc. New York, 2010.
- [4] Shinichi Tamura, Yoshi Nishitani, Takuya Kamimura, Yasushi Yagi, Chie Hosokawa, Tomomitsu Miyoshi, Hajime Sawai, Yuko Mizuno-Matsumoto, Yen-Wei Chen. “Multiplexed Spatiotemporal Communication Model in Artificial Neural Networks,” *Automation, Control and Intelligent Systems*. Vol. 1, No. 6, 2013, pp. 121-130. doi: 10.11648/j.acis.20130106.11.
- [5] S. W. Golomb, G. Gong, *Signal Design for Good Correlation: For Wireless Communication and Rader*, Cambridge University Press. 2005.
- [6] Yoshi Nishitani, Chie Hosokawa, Yuko Mizuno-Matsumoto, Tomomitsu Miyoshi, Hajime Sawai and Shinichi Tamura, “Detection of M-sequences from spike sequence in neuronal networks,” *Computational Intelligence and Neuroscience - Special issue on Computational Intelligence in Biomedical Science and Engineering* Volume 2012, 2012.
- [7] Shinichi Tamura, Yoshi Nishitani, Chie Hosokawa, Yuko Mizuno-Matsumoto, Takuya Kamimura, Yen-Wei Chen, Tomomitsu Miyoshi, Hajime Sawai. “M-sequence family from cultured neural circuits,” *6th International Conference on New Trends in Information Science, Service Science and Data Mining (ISSDM 2012)*, Oct. 23-25, 2012.
- [8] Daniel L. Schacter, Daniel T. Gilbert, Daniel M. Wegner, *Psychology (2nd ed.)*, Worth Publishers, New York, 2011.
- [9] Christophe Lecerf, “The double loop as a model of a learning neural system,” *Proceedings World Multiconference on Systemics, Cybernetics and Informatics*, Vol.1, pp. 587-594, 1998.
- [10] Y. Choe, “Analogical cascade: a theory on the role of the thalamo-cortical loop in brain function,” *Neurocomputing* 52-54, pp.713-719, 2003.
- [11] Takuya Kamimura, Yoshi Nishitani, Yen-Wei Chen, Yasushi Yagi, and Shinichi Tamura, “Copy of neural loop circuits for memory and communication,” *Journal of Communications and Information Sciences*, Vol.4, No.1, pp.46-56, Jan 2014.
- [12] Wulfram Gerstner and Werner Kistler, *Spiking Neuron Models. Single Neurons, Populations, Plasticity*, Cambridge University Press, 2002.
- [13] Daniel Delling, Peter Sanders, Dominik Schultes, Dorothea Wagner, “Engineering Route Planning Algorithms,” in Jürgen Lerner, Dorothea Wagner, Katharina A. Zweig (eds.), *Algorithmics of Large and Complex Networks, Design, Analysis, and Simulation*, 2009.
- [14] Paulsen, O.; Sejnowski, T. J. (2000). "Natural patterns of activity and long-term synaptic plasticity". *Current Opinion in Neurobiology*, 10 (2): 172–179. doi:10.1016/S0959-4388(00)00076-3.
- [15] Steven Yantis, *Sensation and Perception*, Worth Publishers, New York, 2014.